

# THE SOLUTION OF LARGE EFIE PROBLEMS VIA PRECONDITIONED MULTILEVEL FAST MULTIPOLE ALGORITHM

T. Malas<sup>1,2</sup>, L. Gürel<sup>1,2</sup>

<sup>1</sup>Department of Electrical and Electronics Engineering

<sup>2</sup>Computational Electromagnetics Research Center (BiLCEM)

Bilkent University, TR-06800, Bilkent, Ankara, Turkey

E-mail: tmalas@ee.bilkent.edu.tr, lgurel@bilkent.edu.tr

fax: +90-312-2905755

**Keywords:** Preconditioning of the integral equation methods, electric-field integral equation, multilevel fast multipole algorithm, electromagnetic scattering.

## Abstract

We propose an effective preconditioning scheme for the iterative solution of the systems formulated by the electric-field integral equation (EFIE). EFIE is notorious for producing difficult-to-solve systems. Especially, if the target is complex and the utilized frequency is high, it becomes a challenge to solve these dense systems with even robust solvers such as full GMRES. For this purpose, we use an inner-outer solver scheme and use an approximate multilevel fast multipole algorithm for the inner solver to provide a very efficient approximation to the dense linear system matrix. We explore approximation level and inner-solver accuracy to optimize the efficiency of the inner-outer solution scheme. We report the solution of large EFIE systems of several targets to show the effectiveness of the proposed approach.

## 1 Introduction

In this paper we consider fast iterative solutions of the integral equation methods, which yield large and dense linear systems in the form of

$$\bar{Z} \cdot x = b. \quad (1)$$

The solution of such matrix-equations may have prohibitively large computational costs, unless fast methods, such as the multilevel fast multipole algorithm (MLFMA) [4], is employed for the matrix-vector multiplication that is required at least once in an iterative method. If  $N$  denotes the number of unknowns, MLFMA performs the matrix-vector multiplication in  $O(N \log N)$  complexity. Hence, provided that the number of iterations does not grow rapidly as  $N$  grows, integral-equation methods combined with MLFMA provide fast and accurate solutions of large electromagnetic problems.

However, when the target geometry involves open surfaces, the only applicable formulation is the electric-field integral equation (EFIE), which produces ill-conditioned matrices that are difficult to solve iteratively. Particularly, as the geometry size grows in terms of the wavelength, the system matrix becomes nearly singular and it becomes a challenge to solve these large linear systems in moderate memory and CPU time. For this reason, there is strong need for developing parallel preconditioners that can be embedded in a parallel MLFMA implementation [6].

If the preconditioner is constructed from the near-field matrix, such as the sparse approximate inverse preconditioner [2], it lacks the information contained in the far-field interactions, which become dominant for large problems. Hence, preconditioners relying on only the near-field interactions are not sufficiently strong for EFIE problems. As a remedy, we propose an efficient approximation strategy to MLFMA, which is used to build a preconditioner that carries enough information from the far-field interactions. Hence, the preconditioning operation is performed by an iterative solver, which is nested in an outer iterative method used for the solution of (1). The performance of the approximate MLFMA preconditioner is optimized by adjusting the parameters of the MLFMA used in the inner iterations.

We show the effectiveness of the proposed approach by solving a square patch with various sizes. Particularly, we provide the solution of a  $256\lambda \times 256\lambda$  problem that leads to a matrix-equation with 21,965,824 unknowns. This is the largest EFIE problem reported, to the best of our knowledge. The problem is solved on 16 nodes of a cluster with Intel Xeon 5355 processors. We show the accuracy of the solutions of patch problems by comparing them with the physical optics (PO) solutions. We also present the solution of some other targets including a real-life problem.

## 2 Approximate MLFMA Preconditioner

The usual practice in MLFMA is to keep the lowest level cluster-size fixed and partition the target in a bottom up fashion [5]. Because of this, as the problem size and the number of MLFMA levels increase, the near-field matrix becomes more and more sparse. Therefore, for large problems, preconditioners generated from the near-field matrix cannot be strong enough for EFIE and we may need more than what is provided by the near-field matrix. We can make the near-field matrix denser by increasing the size of the lowest-level clusters. However, this is very costly for memory use, which is critical in large-scale simulations. Also, the manipulation of a denser near-field matrix (matrix-vector multiplication or preconditioner generation) can turn out to be unaffordable in terms of CPU time.

On the other hand, we have the opportunity to use an iterative method for preconditioning when we use a flexible solver to solve the linear system (1) [9]. Hence, we can make use of MLFMA to have stronger preconditioners with respect to those obtained from the near-field matrix. This approach produces a nested implementation of iterative solvers [9]. In the outer solver that solves the original system, we use FGMRES, which is a flexible version of GMRES. FGMRES allows the preconditioner to change from iteration to iteration. Then, the preconditioner of this solver can be another Krylov subspace solver which is called the inner solver. We illustrate this preconditioning scheme in Figure 1. The inner solver makes use of an approximate MLFMA (AMLFMA) for efficiency and (possibly) a SAI preconditioner to accelerate its convergence.

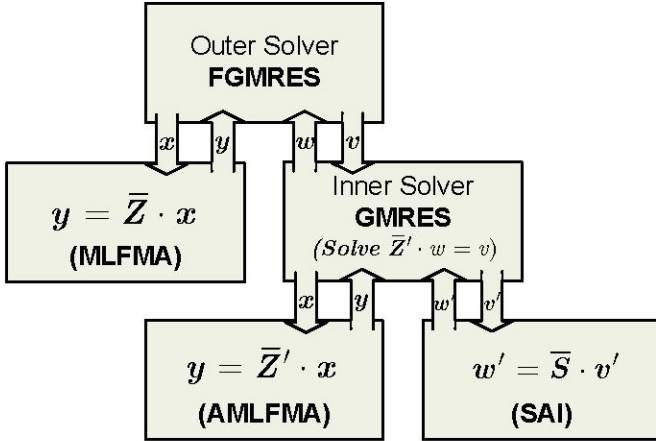


Figure 1: Inner-outer solution scheme.  $\bar{Z}'$  represents the linear operator whose matrix-vector multiplication is provided by AMLFMA.

There can be several ways of achieving cheaper versions of MLFMA. Now we discuss these possibilities and their suitability to use as a preconditioner.

### 2.1 Less Accurate MLFMA Schemes

We control the maximum error of MLFMA by the truncation number

$$L \approx 1.73ka + 2.16(d_0)^{2/3}(ka)^{1/3} \quad (2)$$

of the translation function, where  $a$  is the cluster size of the level and  $d_0$  is the accurate number of digits [8]. We group the relaxation strategies of MLFMA into three:

1) *By Reducing the Number of Accurate Digits:* A less accurate but cheaper version of MLFMA can be constructed by reducing the number of accurate digits  $d_0$  as in [3]. However, the truncation number loosely depends on the value of  $d_0$  for large boxes in the higher levels of MLFMA. For example, for an eight-level problem, if the number of accurate digits is reduced from four to one as in [3], the truncation number of the highest level decreases from 380 to 361, and this corresponds to only 5% reduction. Hence, as the problem size increases, this approach becomes less effective. Moreover, new sets of arrays are needed for the radiation (receiving) patterns of the basis (testing) functions for the less-accurate MLFMA, and this adds a significant cost to the memory requirement.

2) *By Omitting Interactions at High Levels:* Another way to obtain a less-accurate MLFMA can be to interrupt the aggregation process at some level before reaching the top of the tree structure. Then, translation and disaggregation processes are also ignored for highest levels. We call this version incomplete MLFMA (IMLFMA). This approximation scheme requires neither extra computational load during the setup nor significant modifications to the original MLFMA. On the other hand, the processing time required for each level of MLFMA is approximately same, hence, half of the levels should be ignored to obtain 50% reduction in time. This leads to a poor approximation of MLFMA since most of the interactions (usually much larger than the half of the interactions) are not computed. Therefore, IMLFMA usually fails to provide the desired level of accuracy with significant gain from the computational time.

3) *A More Flexible Strategy (AMLFMA):* In order to balance the accuracy and efficiency in a flexible way, we redefine the truncation number for level  $l$  as

$$L'_l = L_1 + a_f(L_l - L_1), \quad (3)$$

where  $L_1$  is the truncation number defined for the first level,  $L_l$  is the original truncation number for the level  $l$  calculated by using (2). The approximation factor  $a_f$  is defined in the range from 0.0 to 1.0. As  $a_f$  increases from 0.0 to 1.0, the AMLFMA becomes more accurate but less efficient, while it corresponds to the full MLFMA when  $a_f = 1$ . Hence, this parameter provides us important flexibility in designing the preconditioner. Moreover, the truncation number of the lowest level is not modified, hence AMLFMA does not require extra computation load for the radiation and receiving



patterns of the basis and testing functions when it is used in conjunction with MLFMA in a nested manner.

We compare the change in truncation numbers and corresponding errors for different approaches in Figures 3 and 4. We note that computational time of the operations for a level are proportional to  $L^2$  [5]. Therefore, we expect significant gains for low approximation factors.

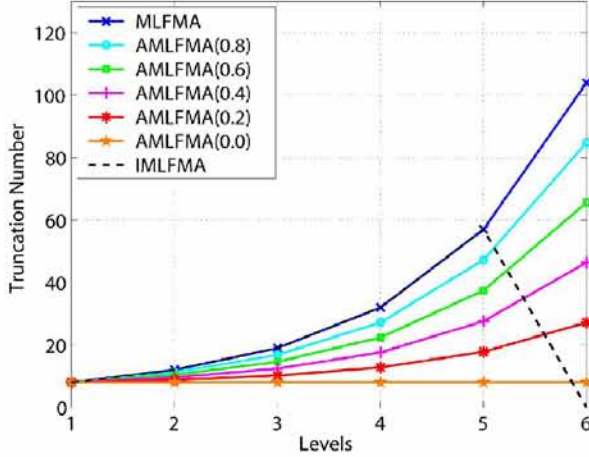


Figure 2: Truncation numbers of MLFMA attained with different approximation strategies. The geometry is a  $20\lambda \times 20\lambda$  patch with 137,792 unknowns.

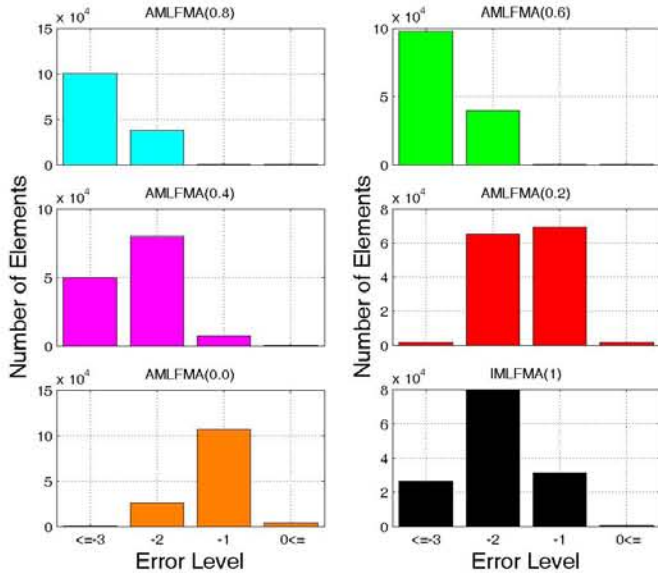


Figure 3: Error levels of various approximations of MLFMA with respect to original MLFMA. IMLFMA(1) is obtained by ignoring interactions of the highest level.

## 2.2 Issues for the Inner-Outer Solution Scheme

There are many factors that effect the performance of inner-outer schemes, such as approximation level of the preconditioning operator to the linear system operator, the choice of the inner solver, inner stopping criteria, and possibly a second preconditioner to be used to accelerate the

inner solver's convergence. Now we discuss these factors in more detail:

*1. Preconditioning operator:* In fact, one can use the same linear system operator for the preconditioning operation by using the same MLFMA for both inner and outer solvers. However, it is known that nesting strategy increases the total number of matrix-vector products with respect to standard Krylov methods [7]. On the other hand, since we only need an approximate solution for preconditioning, a less-accurate MLFMA may increase the efficiency. The discussion in the previous section reveals that the AMLFMA is an appropriate choice. By adjusting the approximation factor  $a_f$  of AMLFMA, both the accuracy and the computational time of the matrix-vector product can be tuned to achieve maximum efficiency.

*2. Inner solver and inner preconditioner:* For the minimization of the overall cost, the preconditioner (i.e., the inner system and inner solver) should provide a satisfactorily accurate solution to a nearby system with possible least effort. However, we observe that satisfaction level of the solution can be quite low, especially for small approximation factors. Hence, for the iterative solver, GMRES seems a good choice, because it provides rapid residual-error decrease in early iterations, providing sufficiently accurate solutions in short times. Also, it can be beneficial to further accelerate the inner solver with a fixed preconditioner. In this context, SAI seems a good candidate since it is successful in reducing the error in early iterations [3].

*3. Inner stopping criteria:* Related to the other choices, the relative residual error and the maximum number of allowable iterations for the inner solver should be carefully selected. In many instances, even the achievement of 0.1 inner error may take many iterations, hence the maximum inner iteration number should be determined carefully to prevent unnecessary work when the iterations stagnate.

Combining the previous discussions, we conclude that SAI preconditioned GMRES targeting 0.1 residual error and using AMLFMA with  $a_f = 0.2$  seems the most appropriate choice. As shown in Figures 2 and 3, for an approximate matrix-vector multiplication with 0.2 incomplete factor, almost all elements of the output vector is computed with less than 0.1 error (with respect to full MLFMA), while the computational time is significantly reduced. Hence, when we fix the target residual error to 0.1,  $a_f = 0.2$  seems the best choice. Lower residual errors necessitate a more accurate matrix-vector multiplication, whose computation time cannot be reduced so effectively.

## 3 Results

In this section we demonstrate the performance of the AMLFMA preconditioner by comparing it with SAI, which is commonly used in integral equation methods [3,11]. In Figure 4, we show the open geometries that we use in our



numerical experiments, i.e., a patch, a half sphere, an open cube with one missing face, and a reflector antenna. These problems are solved for increasing frequencies, which require denser meshes, larger number of unknowns and more MLFMA levels. We note that AMLFMA becomes more efficient with increasing number of levels.

In our experiments we use the GMRES and FGMRES solvers for their robustness. We try to reduce the norm of the initial residual by  $10^{-6}$  in 1,000 iterations, unless stated otherwise. We perform the parallel tests on a 16-node cluster connected via Infiniband network. The nodes have dual Xeon 5355 processors and 16 GB of RAM.

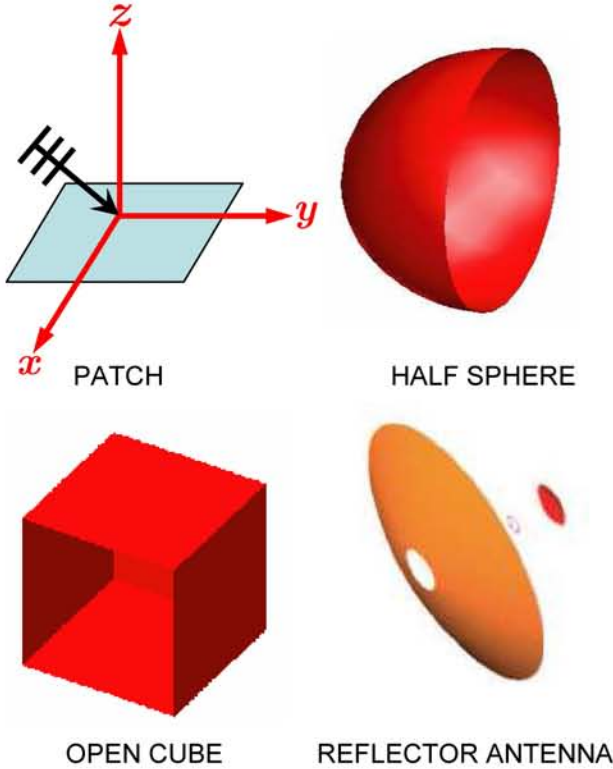


Figure 4: The targets that are used in the numerical experiments. For the half-sphere and the open cube, the illumination is from top. For the reflector antenna, a dipole source is used.

In Table 1 we present the solutions of the patch geometry. For SAI we use the near-field matrix pattern for the approximate inverse. For AMLFMA, we use  $a_f = 0.2$  for and a stopping tolerance of 0.1 or a maximum of 10 iterations for the inner solver. The results in Table 1 indicate that as the problem size increase, the AMLFMA preconditioner becomes more effective compared to SAI. The solution time is halved for large problems. Furthermore, with AMLFMA, we are able to solve an approximately 22-million-unknown problem that corresponds to a  $256\lambda \times 256\lambda$  patch. Since this is a very large problem in terms of  $\lambda$ , we compare the MLFMA solution with the PO solution in Figure 5. The incoming field is a  $y$ -oriented plane wave on the  $xz$ -plane and makes  $45^\circ$  with the  $z$ . We expect PO solution to be particularly accurate for

the specular reflection ( $\phi = 45, \phi = 180$ ) and for forward scattering ( $\phi = 135, \phi = 180$ ). Hence, the accuracy of the MLFMA solution is verified with a perfect agreement between the two methods at these points.

Next we present the solutions of the half sphere and open cube geometries. For both of the problems, there is a very significant decrease in the solution time with AMLFMA preconditioner. The largest problems, cannot be solved with SAI. On the other hand, in reasonable durations the largest problems can be solved with AMLFMA.

$N$	SAI		AMLFMA		
	Iter	Time	Outer	Inner	Time
790,656	12	654	23	222	388
3,164,544	195	5,679	36	354	2,740
12,662,016	275	33,557	53	526	16,184
21,965,824	-	-	9	85	24,689

Table 1: Number of iterations and solution times for the patch geometry. The dash (“-”) denotes that solution cannot be achieved due to memory limitations. The largest problem is solved with  $10^{-3}$  iterative residual accuracy.

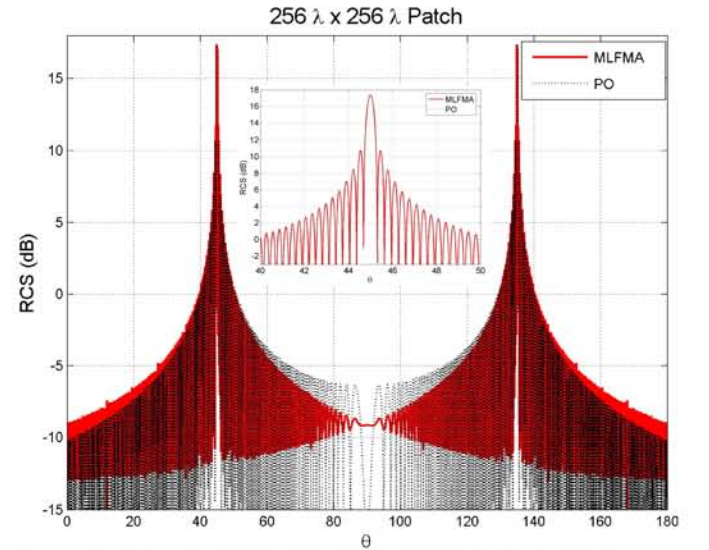


Figure 5: Bistatic RCS with PO and MLFMA for the largest patch geometry having 21,965,824 unknowns.

Finally in Table 4, the results for the reflector antenna are demonstrated. Even though the solution of the smaller problem is achievable with both SAI and AMLFMA, the largest problem again, can only be solved with AMLFMA. Hence, the success of AMLFMA is also shown on a real-life problem.

## 4 Conclusion

In this work, we take advantage of the MLFMA structure to generate a very effective preconditioner. We solve a nearby system approximately but quickly with AMLFMA, and

embed the solution in the main iterative solution. The approximation level of the proposed AMLFMA can be tuned with a parameter, enabling optimization of the global cost of

$N$	SAI		AMLFMA		
	Iter	Time	Outer	Inner	Time
116,596	156	128	21	208	88
2,554,736	547	17,404	70	700	6,621
10,221,280	>2,000	-	44	440	20,774

Table 2: Number of iterations and solution times for the half sphere geometry. The largest problem is solved with  $10^{-3}$  accuracy.

$N$	SAI		AMLFMA		
	Iter	Time	Outer	Inner	Time
65,696	205	58	34	336	57
263,032	404	445	52	517	361
1,052,624	>2,000	-	161	1,607	4,134

Table 3: Number of iterations and solution times for the open cube geometry.

$N$	SAI		AMLFMA		
	Iter	Time	Outer	Inner	Time
356,439	66	124	15	128	104
2,515,103	>1,000	-	322	3,205	25,740

Table 4: Number of iterations and solution times for the reflector antenna.

the solver for maximum performance. We propose a strategy to adjust the approximation level and the inner-solve accuracy, which produces outstanding performance in open geometry problems that has to be modeled with EFIE. Our strategy renders many difficult systems solvable, and for those converging with SAI, the speed up is more than two for the patch and half sphere geometries. Hence, we have been able solve several problems with millions of unknowns in modest iteration counts and solution times.

## Acknowledgements

This work was supported by the Scientific and Technical Research Council of Turkey (TUBITAK) under Research Grant 105E172, by the Turkish Academy of Sciences in the framework of the Young Scientist Award Program (LG/TUBA-GEBIP/2002-1-12), and by contracts from ASELSAN and SSM. Computer time was provided in part by a generous allocation from Intel Corporation.

## References

- [1] S. Balay, W. D. Gropp, L. C. McInnes, and B. F. Smith, "PETSc users manual," Argonne National Laboratory, Tech. Rep. ANL-95/11- Revision 2.1.5, 2004.
- [2] M. Benzi, "Preconditioning techniques for large linear systems: a survey," *J. Comput. Phys.*, vol. 182, no. 2, pp. 418–477, 2002.
- [3] B. Carpentieri, I. S. Duff, L. Giraud, and G. Sylvand, "Combining fast multipole techniques and an approximate inverse preconditioner for large electromagnetism calculations," *SIAM J. Sci. Comput.*, vol. 27, no. 3, pp. 774–792, 2005.
- [4] W. C. Chew, J.-M. Jin, E. Michielssen, and J. Song, Eds., *Fast and Efficient Algorithms in Computational Electromagnetics*. Norwood, MA, USA: Artech House, Inc., 2001.
- [5] Ö. Ergül, "Fast multipole method for the solution of electromagnetic scattering problems," Master's thesis, Bilkent University, Ankara, Turkey, 2003.
- [6] Ö. Ergül and L. Gürel, "Efficient parallelization of multilevel fast multipole algorithm," in *Proc. European Conference on Antennas and Propagation (EuCAP)*, no. 350094, 2006.
- [7] J. van den Eshof, G. L. G. Sleijpen, and M. B. van Gijzen, "Relaxation strategies for nested Krylov methods," *J. Comput. Appl. Math.*, vol. 177, no. 2, pp. 347–365, 2005.
- [8] M. L. Hastriter, S. Ohnuki, and W. C. Chew, "Error control of the translation operator in 3D MLFMA," *Microwave Opt. Technol. Lett.*, vol. 37, no. 3, pp. 184–188, 2003.
- [9] J. Lee, J. Zhang, and C.-C. Lu, "Sparse inverse preconditioning of multilevel fast multipole algorithm for hybrid integral equations in electromagnetics," *IEEE Trans. Antennas and Propagation*, vol. 52, no. 9, pp. 158–175, 2004.
- [10] Y. Saad, *Iterative Methods for Sparse Linear Systems*, 2nd ed. Philadelphia, USA: SIAM, 2003.
- [11] V. Simoncini and D. B. Szyld, "Flexible inner-outer Krylov subspace methods," *SIAM J. Numer. Anal.*, vol. 40, no. 6, pp. 2219–2239, 2002.